

Amendments to the Specification:

Please replace paragraph starting at page 25, line 26 with the following amended paragraph:

-- In order to benefit from the efficiency of the duplex (two identical systems in parallel executing the same software with a comparison of the outputs) that is a means of detecting all errors without exceptions regardless of their type (data error, address error, sequencing error, configuration error, etc.) while eliminating structural redundancy, the process according to the invention consists of installing a duplex operation on a single physical ~~sequence~~ channel. In a given real time cycle, two virtual ~~sequences~~ channels located on the same physical ~~sequence~~ channel are multiplexed in time; the data generated from each execution of a virtual ~~sequence~~ channel are stored in “time multiplexed duplex tables” (for example commands, context) so that they can be voted before use. --

Please replace paragraph starting at page 26, line 28 with the following amended paragraph:

-- Actions on the output side of the vote module, in other words firstly data transfers to the control electronics (i.e. the data bus) and secondly the control electronics itself, are not protected by the process according to the invention. The user makes a system analysis to determine these critical commands that need to be error-free and to protect them by mechanisms well known to an expert in the subject; coding of data, self-checking circuit, instrumentation of the control electronics, etc. --

Please replace paragraph starting at page 31, line 25 with the following amended paragraph:

-- Thus for example for virtual sequence 1, during cycle #N in task K, the first table may be considered as "New" and is denoted TAB-Ctxt-New #1, the second table is ~~"old"~~ "Old" and is denoted TAB-Ctxt-Old #1. If the vote for task K does not detect any errors, the swap inverts the roles at the end of the vote; the first table then becomes TAB-Ctxt-Old #1, and the second table becomes TAB-Ctxt-New #1. --

Please replace paragraph starting at page 34, line 15 with the following amended paragraph:

-- - swap the context tables by changing the index: TAB-Ctxt-New replaces ~~TAB-Ctxt-old~~ TAB-Ctxt-Old and is used as the context for the next real time cycle, --

Please replace paragraph starting at page 34, line 25 with the following amended paragraph:

-- Thus in FIGURE 9, during the real time cycle N+1, if no errors were detected during the real time cycle N, the entry context to task A is ~~TAB-Ctxt-Old(N)~~ TAB-Ctxt-Old(N), this table actually containing the data from TAB-Ctxt-New(N) due to the swap; if errors were detected, the entry context to task A is TAB-Ctxt-Old(N-1), this table being identical to table TAB-Ctxt-Old(N-1) in the real time cycle N since context switching does not take place in the case of an error. --

Please replace paragraph starting at page 37, line 4 with the following amended paragraph:

-- a) check the state of the microprocessor connected at the beginning of the vote and the state of the control unit module: check that the stack pointer is within the authorized area, check the microprocessor and the control unit card/board configuration registers; --

Please replace paragraph starting at page 37, line 16 with the following amended paragraph:

-- d) activate the key for the memory access ~~entering~~ monitoring device indicating that voting is being done and authorizing simultaneous access to the two memory areas ChV#1 and ChV#2; --

Please replace paragraph starting at page 38, line 13 with the following amended paragraph:

-- n) Swap the context tables for task K by inverting the pair of "~~old~~" "Old" and "New" indexes stored in memory; --

Please replace paragraph starting at page 48, line 12 with the following amended paragraph:

-- ~~Hardware~~ Software embodiment --